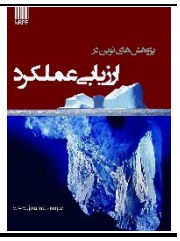
Modern Research in Performance Evaluation

www.journal-mrpe.ir

Mod. Res. Perf. Eval. Vol. 1, No. 3 (2022) 219–225.



Paper Type: Original Article

Evaluating Operations Research Software: A Method Based on Data Envelopment Analysis

Mahnaz Maghbouli[1,*], Azam Pourhabib Yekta[2]

[1] *Islamic Azad University, Aras Branch, Jolfa, Iran; mmaghbouli@gmail. com.*
[2] *Islamic Azad University, Sowmesara Branch, Guilan, Iran; apourhabibyekta@yahoo.com.*

Citation:

Maghbouli, M., & Pourhabib Yekta, A. (2022). Evaluating operations research software: a method based on data envelopment analysis. *Modern research in performance evaluation, 1*(3), 219-225.

Received: 16/03/2022 Reviewed: 12/04/2022 Revised: 09/05/2022 Accepted: 01/07/2022

Abstract

Propose: Operation Research (OR) has attracted the interest of many researchers. As human activity increases, the use of computers in implementing accurate and fast optimization algorithms seems inevitable. Usability and reliability are two important attributes for software quality and can play an important role in improving user experience.

Methodology: In this paper, usability is presented as the user's effort in performing a task in the software application and reliability is presented as the ability to run a program in a given environment in a given period of time without errors. These two criteria are measured using a standard questionnaire. Five common software programs OR are evaluated. A random sample was used to select 20 questionnaires and determine the usability and reliability scores. Data Envelopment Analysis (DEA) models without inputs were used to evaluate the five selected programs.

Findings: The results indicate that the software GAMS was assigned the first role in the evaluation of five common OR softwares.

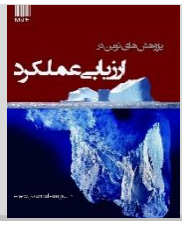
Originality/Value: Software evaluation can play an important role in any field from business and industry to entertainment and entertainment.

Keywords: Data envelopment analysis, Performance evaluation, Reliability, Software, Usability.

Corresponding Author: mmaghbouli@gmail.com http://dorl.net/dor/20.1001.1.28211960.1401.1.3.6.2

Licensee. **Modern Research in Performance Evaluation**. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0).

پژوهش‌های نوین در ارزیابی عملکرد
دوره ۱، شماره (۳)، (۱۴۰۱)، ۲۲۵-۲۱۹
www.journal-mrpe.ir
Modern Research in Performance Evaluation



نوع مقاله: پژوهشی

ارزیابی نرم‌افزارهای تحقیق در عملیات: روشی مبتنی بر تحلیل پوششی داده‌ها

مهناز مقبولی[۱*]، اعظم پورحبیب یکتا [۲]
[۱]دانشگاه آزاد اسلامی، واحد ارس، آذربایجان شرقی، ایران.
[۲]دانشگاه آزاد اسلامی، واحد صومعه‌سرا، گیلان، ایران.

چکیده

هدف: تحقیق در عملیات توجه بسیاری از محققین را به خود جلب کرده است. با افزایش فعالیت انسان، استفاده از رایانه در پیاده‌سازی الگوریتم‌های بهینه‌سازی دقیق و سریع اجتناب ناپذیر به‌نظر می‌رسد. قابلیت استفاده و قابلیت اطمینان دو ویژگی مهم برای کیفیت نرم‌افزار هستند و می‌توانند نقش مهمی در بهبود تجربه کاربر داشته باشند.

روش‌شناسی پژوهش: در این مقاله قابلیت استفاده به‌عنوان تلاش کاربر در انجام یک کار در نرم‌افزار کاربردی و قابلیت اطمینان به‌عنوان توانایی اجرای یک برنامه در یک محیط معین در یک بازه زمانی معین بدون خطا ارائه شده است. این دو معیار با استفاده از پرسشنامه استاندارد اندازه‌گیری می‌شوند. پنج نرم افزار معمولی در تحقیق در عملیات ارزیابی می‌شوند. برای انتخاب ۲۰ پرسشنامه و تعیین امتیاز قابلیت استفاده و پایایی از نمونه تصادفی استفاده شده است. برای ارزیابی پنج برنامه انتخاب‌شده از مدل‌های تحلیل پوششی داده‌ها (*DEA*) بدون ورودی استفاده شده است.

یافته ها: نتایج حاکی از آن است که نرم افزار *GAMS* در ارزیابی پنج نرم افزار رایج در تحقیق در عملیات نقش اول را به خود اختصاص داده است.

اصالت/ارزش افزوده: ارزیابی نرم‌افزار می‌تواند نقش مهمی در هر زمینه‌ای از تجارت، صنعت و سرگرمی داشته باشد.

کلیدواژه‌ها: قابلیت اطمینان، قابلیت استفاده، تحلیل پوششی داده‌ها، ارزیابی عملکرد، نرم‌افزار.

۱- مقدمه

امروزه، نرم‌افزار به‌عنوان زمینه‌ای آشنا و فراگیر در تمامی جنبه‌های زندگی روزمره، از کار و تجارت تا تفریحات گسترش‌یافته است. فراگیر شدن نرم‌افزار منجر به افزایش تعداد و تنوع کاربران آن شده است. امروزه نرم‌افزارها، نقش مهمی در سیستم‌هایی با کاربردهای حساس همچون پزشکی، صنایع نظامی و راکتورهای هسته‌ای پیداکرده‌اند. قابلیت استفاده نرم‌افزار که در فارسی معادل «استفاده‌پذیری» و «کاربردپذیری» ترجمه‌شده است، یکی از ویژگی‌های کیفیت نرم‌افزار است که در آن استفاده کاربران از نرم‌افزار برای رسیدن به اهداف خود به‌صورت اثربخش، کارا و با رضایت مدنظر قرار می‌گیرد. قابلیت استفاده، یکی از ویژگی‌های مهم در مقبول و فراگیر شدن نرم‌افزار در میان

نویسنده مسئول
mmaghbouli@gmail.com
http://dorl.net/dor/20.1001.1.28211960.1401.1.3.6.2

کاربران آن و برتری نسبت به نرم‌افزارهای مشابه است (مازنی و وفادار[1]، ۲۰۱۸). سهولت استفاده به رضایت عمومی از نرم‌افزار، به‌ویژه ازنظر واسط کاربری بازمی‌گردد. معیار مهم دیگری که در بررسی نرم‌افزارها موردتوجه قرار می‌گیرد تعیین کمی میزان قابلیت اعتماد نرم‌افزار است. قابلیت اعتماد نرم‌افزار احتمال عملکرد بدون شکست یک برنامه کامپیوتری در محیطی مشخص و برای زمانی معین تعریف می‌شود. باید توجه داشت که سیستم کامپیوتری شامل دو مولفه سخت‌افزار و نرم‌افزار است؛ اما نکته قابل‌توجه این است که سیستم نرم‌افزاری درواقع به معنای فیزیکی خراب نمی‌شود (اختریان و یعقوبی[2]، ۲۰۱۵). خطاهای نرم‌افزاری بیشتر به دلیل اشتباهات ذهنی سازندگان است و سالخوردگی و استهلاک در نرم‌افزار موجود نیست. حال آن‌که در بخش سخت‌افزاری استهلاک به معنای فیزیکی آن روی می‌دهد. قابلیت اطمینان یک نرم‌افزار به فرآیند کشف خطاها و حذف آن‌ها بستگی دارد. در مدل‌سازی قابلیت اطمینان نرم‌افزار فرض بر این است که وقوع شکست‌ها و عیب‌یابی آن‌ها کاملا تصادفی هستند. بدین‌صورت که نرم‌افزار می‌تواند به‌عنوان تابعی در نظر گرفته شود که داده‌های ورودی را به خروجی تبدیل می‌کند؛ اما دراین‌بین، خطا در واردکردن ورودی‌ها امری بدیهی است و به دنبال آن شکست نیز به‌صورت تصادفی رخ می‌دهد. لذا پژوهشگران با اتکا به مدل‌های ریاضی درصدد کشف و کاهش این خطاها و درنتیجه بهبود نرم‌افزار برمی‌آیند. قابلیت اعتماد نرم‌افزار به کاربران نیز کمک می‌کند تا با اطمینان کافی برای خرید نرم‌افزار هزینه کنند. با توجه به گسترش روزافزون و توسعه نرم‌افزارها در همه زمینه‌ها، علوم پایه نیز از این جهش بی‌بهره نمانده و نرم‌افزارهای متعددی در خدمت علوم پایه به‌خصوص علم تحقیق در عملیات قرارگرفته‌اند. هرکدام از این نرم‌افزارها باقابلیت‌های متعدد، تنوع و رقابتی را در بازار فراهم آورده‌اند؛ اما آنچه در این سال‌ها بیشتر مورداستفاده و نظر بوده است، دو مورد قابلیت استفاده و قابلیت اطمینان نرم‌افزارهای موجود در بازار است. ازاین‌رو مساله ارزیابی قابلیت استفاده و قابلیت اطمینان نرم‌افزارهای موجود به یکی از مسایل مطرح در صنعت نرم‌افزار تبدیل‌شده است؛ و همین امر سبب ایجاد حوزه تحقیقاتی جالبی برای پژوهشگران شده است (تهرانی و جمشیدی[3]، ۲۰۱٤؛ مازنی و وفادار ۲۰۱۸؛ عبدالعزیز و همکاران[4]، ۲۰۱۶؛ جوریستو و همکاران[5]، ۲۰۱۷؛ هوانگ و همکاران[6]، ۲۰۲۲، کومار و همکاران[7]، ۲۰۲۲؛ لین و همکاران[8]،۲۰۲۲). در پژوهش اول، با استفاده از پرسشنامه نظرات کاربران در ارتباط با سادگی استفاده، رضایت و مفید بودن کارکردها مورد ارزیابی قرار می‌گیرد. پژوهش دوم، استفاده صرف از پرسشنامه را مورد چالش قرار می‌دهد و لذا دو روش مختلف ارزیابی «سهولت استفاده» انتخاب و نتایج حاصل از این دو روش با یکدیگر مقایسه شده است. ارزیابی با استفاده از پرسشنامه و ارزیابی بر اساس عملکرد کاربران در استفاده از نرم‌افزار زیرمعیارهایی مانند تعداد گام‌های کمتر برای کامل کردن وظایف، تعداد خطای کمتر، اصلاح خطا به‌صورت سریع و آسان، نیاز به تلاش و زمان کمتر و استفاده موفقیت‌آمیز کارکردهای نرم‌افزار در نظر گرفته می‌شوند و بدین ترتیب، امکان مقایسه نتایج حاصل از دو روش ارزیابی فراهم می‌گردد. عبدالعزیز و همکاران در مقاله خود با عنوان «روشی برای بهبود قابلیت استفاده در محصولات نرم‌افزاری» روشی برای پر کردن خلا بین سیستم و کاربر در جهت موثرتر و کارآمدتر کردن قابلیت استفاده از نرم‌افزار را پیشنهاد می‌دهد. در این مقاله، تکنیک نمونه‌سازی بر مبنای سادگی، مورداستفاده نویسندگان قرار است. در مقاله جوریستو و همکاران، نشان داده‌شده است که بین طراحی نرم‌افزار و قابلیت استفاده رابطه مستقیم وجود دارد. همچنین در این مقاله، تعریفی از تاثیر ادغام کردن ویژگی «قابلیت استفاده خاص» در لحظه طراحی نرم‌افزار است که بر اساس تعدادی از نرم‌افزارهای واقعی نتیجه گرفته شده است. هوانگ و همکاران (۲۰۲۲) اشاره به ناکامل بودن فرآیند حذف خطاها می‌کند و مطالعه را بر اساس توزیع ناهمگن پوآسن با در نظر گرفتن عدم پایداری و ثبات حذف خطاها، تنوع خطاها و نقاط تغییر در طول مدت انجام آزمایش جهت گسترش مدل‌های رشد قابلیت اطمینان نرم‌افزار بنا می‌نهد. ازجمله مطالعات داخلی که درزمینه قابلیت اطمینان نرم‌افزار می‌توان به آن اشاره کرد، مقاله اختریان و یعقوبی (۲۰۱۵) است. در این مقاله با لحاظ کردن وابستگی بین خطاهای پیچیده و نیز تاخیر زمانی بین کشف و حذف خطاهای پیچیده مدل تعمیم‌یافته‌ای بر اساس فرآیند پوآسن ناهمگن ارایه می‌شود که اطلاعات دقیق‌تری در مورد کیفیت نرم‌افزار به دست می‌دهد. لین و همکاران (۲۰۲۲)، به مساله اولویت‌بندی خطاها و حذف خطاهای با اولویت بالاتر و بیشتر اشاره می‌کند و مدلی موسوم به مدل اولویت‌بندی یا همان *PPQ*، پیشنهاد می‌دهد. در این مدل، خطاهای با اولویت بالاتر به‌صورت پیشگیرانه، منابعی که قبلا در اختیار خطاهای با اولویت پایین‌تر بوده است را در اختیار می‌گیرند. مثال‌های عددی متعددی نیز با داده‌های خطای واقعی در دودسته نرم‌افزار *open-source* و *closed-source* به‌تفصیل مورد تحلیل و ارزیابی قرارگرفته است. نتایج حاکی از کاربرد بهتر و دقیق‌تر روش مورداستفاده در مقایسه با مدل‌های موجود رشد قابلیت اطمینان نرم‌افزار است. ایوانف و همکاران[9] (۲۰۱۸) در مقاله خود با عنوان «مقایسه قابلیت اطمینان سیستم‌های نرم‌افزاری: مطالعه موردی در سیستم‌عامل‌های موبایل» از روش *GQM* و

[1] Mazni and Vafadar
[2] Akhtarian and Yaqoubi
[3] Tehrani and Jamshidi
[4] Abdelaziz et al.
[5] Juristo et al.
[6] Huang et al.
[7] Kumar et al.
[8] Lin et al.
[9] Ivanov et al.

مدل‌های توسعه قابلیت اطمینان نرم‌افزار استفاده می‌کند. روش پیشنهادی برای ارزیابی و مقایسه روی سه سیستم‌عامل موبایل مورداستفاده قرار گرفته است. کومار و همکاران (۲۰۲۲) در مقاله خود با نقض اصل «احتمال انتقال حالت» در زنجیره مارکوف، قابلیت اطمینان را در صنایع مختلف و بر اساس روش تصادفی تعمیم‌یافته شبکه‌های پتری، پیش‌بینی نمودند. روش پیشنهادی در سیستم‌های ایمنی-حیاتی و سیستم‌های مورداستفاده در صنایع سنگین و رآکتورها کاربرد فراوانی دارد. قابلیت اطمینان پیش‌بینی‌شده با استفاده از این روش بسیار نزدیک به صد درصد است و تحلیل حساسیت، تحت شرایط و پارامترهای مشخص و معین نیز این رقم را تایید می‌کند. آروانیتو و همکاران[1] (۲۰۲۱)، در مطالعه‌ای جهت توسعه نرم‌افزار و سهولت استفاده از آن به این نتیجه اصلی اشاره می‌کند که توسعه‌دهندگان حرفه‌ای نرم‌افزار بیشتر بر فعالیت‌های افزایش بهره‌وری نظیر استفاده مجدد از کدها و غیره تاکید خاصی دارند. یکی دیگر از یافته‌های کلیدی این مطالعه، کمبود شواهد در مبادلات تمرینی نرم‌افزار یا بررسی تاثیرات غیرمستقیم بر سایر وجوه کیفیت نرم‌افزار است. پایوا و همکاران[2] (۲۰۲۱)، در مطالعه‌ای جامع به مفهوم دسترسی‌پذیری در پروسه‌های مهندسی نرم‌افزار پرداخته‌اند. در این مقاله که تاکید بر آشنایی طراحان و توسعه‌دهندگان نرم‌افزار با روش و ابزارهای جدید دارد، نویسندگان ضمن تاکید بر هم‌پوشانی دو مساله توسعه و «سهولت استفاده نرم‌افزار»، یک روش سیستماتیک جدید در بازه زمانی (۲۰۱۱) تا (۲۰۱۹) را ارایه می‌کنند. نتیجه مطالعه در این بازه زمانی دلالت بر این مطلب دارد که «سهولت استفاده» به‌عنوان تنها ویژگی نهایی نرم‌افزار طبقه‌بندی نمی‌گردد و در طی دوره زمانی استفاده نرم‌افزار معنی‌دار است. انتخاب معیارهای مناسب برای اندازه‌گیری دو ویژگی «سهولت استفاده» و «قابلیت اطمینان» بسیار مهم است. در این پژوهش، قابلیت استفاده نرم‌افزار به معنای درجه و اندازه‌ای است که یک محصول یا سیستم نرم‌افزاری ر در یک حوزه خاص می‌تواند توسط کاربرانی مشخص برای رسیدن به اهداف به‌صورت اثربخش، کارا و با رضایت مورداستفاده قرار گیرد؛ بنابراین سهولت استفاده یا همان قابلیت استفاده نیازمند انتخاب معیارهای مناسبی است تا بتواند به پرسش موردنظر این پژوهش پاسخ دهد. زیر معیارهای موردنظر در این مطالعه عبارت‌اند از قابل‌فهم بودن، قابلیت یادگیری، عملکرد و جذابیت. قابلیت اطمینان نیز بر تعریف وقوع خرابی و عیب بنا شده است. اطمینان‌پذیری یا قابلیت اطمینان درصد کامیابی در انجام دستور کار است، یا درصد اینکه سامانه‌ای بی‌آنکه دچار نارسایی شود به دستور کار برگزیده با مرزهای گزینش‌شده در برنامه‌ریزی (مانند مرزهای زمانی و مکانی) و در بستر کارکردی ویژه (مانند دما، نم، لرزش و غیره) کار کند. زیر معیارهایی نظیر بلوغ، تحمل خطا و قابلیت بازیافت ازجمله مواردی است که در این تحقیق موردتوجه است. نرم‌افزارهای تحقیق در عملیات به‌عنوان پشتیبان‌های اساسی در برنامه‌ریزی و مدل‌سازی از اهمیت ویژه‌ای برخوردارند. نرم‌افزار مناسب تحقیق در عملیات باید از ارتباط استاندارد بین ماژول‌ها استفاده کند. همچنین در خصوص تایید داده‌های جدید به کاربر کمک کرده و به‌صورت آسان اطلاعات قدیمی را برای تجزیه‌وتحلیل روند استخراج کند. البته ویژگی‌های اصلی دیگری همچون انعطاف‌پذیری، سطوح مختلف امنیت و پشتیبان گیری خودکار وجود دارند. تا جایی که مطلع هستیم تنها به معرفی این نرم‌افزارها در کتب و سایت‌های آموزشی بسنده شده و مقاله‌ای در خصوص ارزیابی این نرم‌افزارها و مقایسه آن‌ها با یکدیگر منتشر نشده است. در این مطالعه، هدف، ارزیابی و بررسی عملکرد چند نرم‌افزار موجود و مورداستفاده در درس تحقیق در عملیات است. برای این منظور، دو ویژگی قابلیت استفاده و قابلیت اطمینان با استفاده از ابزار پرسشنامه سنجیده می‌شود. سپس با استخراج نتایج حاصل از این پرسشنامه‌ها، با اتکا به روش‌های نا پارامتریک ازجمله تحلیل پوششی داده‌ها، نتایج مورد ارزیابی قرار می‌گیرد و نرم‌افزار کارآمدتر انتخاب می‌گردد. نتایج این تحقیق، از دو دیدگاه کاربردی و پژوهشی قابل‌توجه است. از دیدگاه کاربردی، این پژوهش می‌تواند مورداستفاده مدرسان و تولیدکنندگان نرم‌افزار قرار بگیرد. طرف دیگر، این پژوهش از جنبه ارزیابی و رتبه‌بندی نیز حایز اهمیت است زیرا نرم‌افزارهای مورداستفاده از دو جنبه سهولت استفاده و قابلیت اطمینان مورداستفاده قرار می‌گیرد. این مقاله به‌صورت زیر ساختاربندی شده است. در بخش دوم به‌صورت مختصر به تحلیل پوششی داده‌ها پرداخته می‌شود. در بخش سوم، نرم‌افزارهای موردنظر در تحقیق در عملیات که در این مطالعه به‌کاررفته معرفی‌شده و نتایج حاصل از ارزیابی و رتبه‌بندی این نرم‌افزارها ارایه می‌شود. مقاله با نتیجه‌گیری و پیشنهادها خاتمه می‌یابد.

۲-مرور ادبیات

تحلیل پوششی داده‌ها[3](*DEA*) یکی از متداول‌ترین روش‌های غیر پارامتری برای ارزیابی عملکرد نهادها و دیگر فعالیت‌های رایج در زمینه‌های مختلف است .علت مقبولیت گسترده این روش نسبت به سایر روش‌ها امکان بررسی روابط پیچیده و اغلب نامعلوم بین چندین ورودی و چندین خروجی است .به‌عبارت‌دیگر، در این روش برنامه‌ریزی ریاضی، واحدهای تصمیم‌گیرنده یا همان *DMUs*[4]

[1] Arvanitou et al.
[2] Paiva et al.
[3] Data Envelopment Analysis (DEA)
[4] Decision Making Unitss (DMUs)

موجودیتی است که ورودی‌های نیمه مثبت چندگانه را به خروجی‌های نیمه مثبت چندگانه تبدیل می‌کنند و ارزیابی آن موردنظر است. اولین مدل سنجش و ارزیابی عملکرد توسط چارنز و همکاران[1] (۱۹۷۸) ارایه گردید و موسوم به مدل CCR[2] است (حسین‌زاده و همکاران[3]، ۲۰۱۶؛ چارنز و همکاران، ۱۹۷۸؛ کوپر و همکاران[4]، ۲۰۰۶). این مدل در دو قالب ورودی محور و خروجی محور مورداستفاده قرار می‌گیرد. فرض کنید n واحد تصمیم‌گیرنده $DMU_j(=1,...,n)$ داریم که هرکدام m ورودی نام نفی $x=(x_1,...,x_m)$ را برای تولید s خروجی $y=(y_1,...,y_s)$ مصرف می‌کند. مدل پوششی CCR در قالب ورودی محور به‌صورت برنامه‌ریزی خطی زیر تعریف می‌گردد (حسین‌زاده و همکاران، ۲۰۱۶؛ کوپر و همکاران، ۲۰۰۶):

$$\begin{aligned} &Min \quad \theta \\ &s,t. \quad \sum_{j=1}^{n}\lambda_j x_{ij} \le \theta x_{io}, \quad i=1,...,m, \\ &\quad\quad \sum_{j=1}^{n}\lambda_j y_{rj} \ge y_{ro}, \quad r=1,...,s, \\ &\quad\quad \lambda_j \ge 0, \quad j=1,...,n. \end{aligned} \qquad (۱)$$

در **مدل (۱)**، مقادیر سمت راست x_{io} و y_{ro} به ترتیب ورودی و خروجی واحد تحت ارزیابی است و پارامتر θ ضریب انقباض ورودی است. متغیر نام نفی λ_j نیز به‌عنوان سهم هر واحد در ارزیابی واحدهای متجانس در مدل هست. اگر جواب بهینه **مدل (۱)** برابر یك ارزیابی گردد واحد تحت ارزیابی را کارا می‌نامند در غیر این صورت آن واحد ناکارا قلمداد می‌گردد. **مدل (۱)** را مدل با بازده به مقیاس ثابت نیز می‌نامند. یکی از توسعه‌هایی که بر مدل CCR تعریف شده است، مدل با بازده به مقیاس متغیر است که توسط بنکر و همکاران[5] (۱۹۸۴) توسعه یافت. مدل موسوم به BCC در فرم پوششی در قالب مساله برنامه‌ریزی خطی تعریف می‌گردد:

$$\begin{aligned} &Min \quad \theta \\ &s,t. \quad \sum_{j=1}^{n}\lambda_j x_{ij} \le \theta x_{io}, \quad i=1,...,m, \\ &\quad\quad \sum_{j=1}^{n}\lambda_j y_{rj} \ge y_{ro}, \quad r=1,...,s, \\ &\quad\quad \sum_{j=1}^{n}\lambda_j = 1, \\ &\quad\quad \lambda_j \ge 0, \quad j=1,...,n. \end{aligned} \qquad (۲)$$

تنها تفاوت **مدل (۲)** با **مدل (۱)** در افزودن محدودیت $\sum_{j=1}^{n}\lambda_j=1$ است (بنکر و همکاران، ۱۹۸٤). مقادیر سمت راست همان مقادیر ورودی و خروجی واحد تحت ارزیابی است. همانند قبل، واحدی کارا ارزیابی می‌شود که در بهینگی مقدار تابع هدف برابر یک و در غیر این صورت واحد ناکارا قلمداد می‌شود. البته در برخی از موارد همانند ارزیابی عملکرد دانش آموزان یا ارزیابی قوت اقتصادی کشورها، ورودی‌ها در این ارزیابی دخالت چندانی ندارد و خروجی‌ها نقش عمده‌ای را بازی می‌کنند. لیو و همکاران[6] (۲۰۱۱) بر تکیه‌بر اصول موضوعه تحلیل پوششی داده‌ها، مدل‌های شعاعی بدون ورودی را پیشنهاد دادند.

در **مدل (۳)** متغیرهای ورودی در ساختار مدل دخالتی ندارند و متغیر نام نفی λ_j نیز به‌عنوان سهم هر خروجی در ارزیابی واحدها در مدل هست. مدل SBM در حالتی که متغیرهای ورودی در مدل دخالتی ندارند به‌صورت زیر پیشنهاد می‌گردد (کای و وو[7]، ۲۰۰۱).

---

[1] Charnes et al.
[2] Charnes, Cooper and Rhodes (CCR)
[3] Hosseinzadeh et al.
[4] Cooper et al.
[5] Banker et al.
[6] Liu et al.
[7] Cai and Wu

$$\begin{aligned} \theta^{*} &= Max \quad \theta \\ s,t. \quad & \sum_{j=1}^{n} \lambda_j y_{rj} \geq \theta y_{ro}, \quad r = 1,...,s, \\ & \sum_{j=1}^{n} \lambda_j = 1, \\ & \lambda_j \geq 0, \quad j = 1,...,n. \end{aligned} \qquad (۳)$$

کای و وو (۲۰۰۱) برای ارزیابی موقعیت مالی یازده شرکت *IT* و با در نظر گرفتن چهار اندیس خروجی از **مدل (۴)** استفاده کردند. هرچند در مقاله آن‌ها دلیل کارکرد این مدل در حالت خاصی که موردبررسی قرار داده بودند ذکر نشده است. یکی از نقاط قوت این مدل کاربرد وسیع آن در استفاده از داده‌های منفی است. برای اطلاعات بیشتر به مقاله لیو و همکاران (۲۰۱۱) رجوع شود.

۳- یافته‌های تحقیق

نرم‌افزارهای مورداستفاده در تحقیق در عملیات بسیار متنوع هستند و کاربران بنا به موارد نیاز و سهولت دسترسی به هرکدام از این نرم‌افزارها، از آن‌ها استفاده می‌کنند. طی مطالعه میدانی صورت گرفته در میان دانش‌آموختگان تحقیق در عملیات، پنج نرم‌افزار مورداستفاده به نام *GAMS*، لیندو، لینگو، تورا و *WinQSB* بیشتر از همه مورداستفاده و نظر بوده است. نخستین گام در این پژوهش، تعیین معیارهای ارزیابی است. دو معیار «سهولت استفاده» و «قابلیت اطمینان» دو معیار موردنظر در این تحقیق هستند. سهولت استفاده اغلب یک مفهوم شهودی است که به‌طور مستقیم قابل ارزیابی نیست و لازم است معیارهای متناظر با آن‌که قابلیت اندازه‌گیری کمی دارند، انتخاب شوند. چهار زیر معیار قابل‌فهم بودن، قابلیت یادگیری، عملکرد و جذابیت برای اندازه‌گیری این معیار در نظر گرفته می‌شود. قابلیت اطمینان درواقع احتمال عملکرد رضایت‌بخش سیستم تحت شرایط کار مشخص در مدت‌زمان معین است. یک شاخص کلاسیک برای ارزیابی قابلیت اطمینان اندازه‌گیری احتمال وقوع یا عدم وقوع ازکارافتادگی سیستم است. از طرفی، دو بحث مطرح در ارزیابی قابلیت اطمینان، خطر و ریسک است. خطر همان تقسیم‌بندی حوادث بر اساس نتیجه حادثه است و ریسک درواقع شدت خطر (احتمال وقوع خطر) است و همواره تلاش برای کاهش ریسک است؛ زیرا خطر به‌خودی‌خود در تمام سیستم‌ها (چه ماموریت محور و چه کارکرد مداوم) وجود دارد و صفر نمی‌شود. همچنین، در این ارزیابی، سه مولفه بلوغ، تحمل خطا و قابلیت بازیافت سه زیر مولفه‌ای هستند که جهت ارزیابی معیار «قابلیت اطمینان» انتخاب شده‌اند. نکته قابل‌توجه اینکه برای ارزیابی کمی قابلیت اطمینان سیستم، باید از اطلاعات آماری گذشته نیز استفاده کرد و درواقع عملکرد گذشته سیستم را نیز ارزیابی نمود. با توزیع پرسشنامه استاندارد در زمینه ارزیابی نرم‌افزار، نمونه‌ای تصادفی متشکل از بیست نفر انتخاب شد و دو معیار موردنظر برای هر پنج نرم‌افزار به‌صورت کمی استخراج گردیده است. هر نرم‌افزار را به‌عنوان واحد تصمیم‌گیرنده یا همان *DMU* انتخاب کردیم. جهت ارزیابی این پنج نرم‌افزار با استفاده از مدل‌های استاندارد تحلیل پوششی داده‌ها، نیاز به معین نمودن ورودی‌ها و خروجی‌هاست. با توجه به استخراج دو معیار «سهولت استفاده» و «قابلیت اطمینان» از طریق پرسشنامه‌ها، این دو مولفه به‌عنوان خروجی در نظر گرفته‌شده است. دلیل انتخاب این مولفه‌ها به‌عنوان خروجی این است که مطابق با تعاریف رایج در تحلیل پوششی داده‌ها، خروجی عاملی هست که افزایش آن سبب افزایش کارایی واحد تصمیم‌گیرنده می‌شود. ازاین‌رو، این دو معیار به‌عنوان خروجی انتخاب شده‌اند. نکته قابل تامل نیز این است که در هنگام استفاده از نرم‌افزارها، لینک‌ها، کلمات و حتی اشتباهات تایپی به‌عنوان داده ورودی محسوب می‌شوند؛ اما تعداد خطای کمتر و یا اصلاح خطا به‌صورت سریع و آسان که از مولفه‌های کارکرد موفقیت‌آمیز نرم‌افزار هستند در این مطالعه وارد نشده است. لذا مدل‌های موردنظر برای ارزیابی، مدل‌های تحلیل پوششی داده‌ها بدون ورودی است. میانگین معیارهای استخراج‌شده برای هرکدام از بیست پرسشنامه توزیع‌شده به‌عنوان معیار کمی مورداستفاده در مدل‌های موردنظر، خواهد بود. این اعداد در **جدول ۱** آورده شده است.

جدول ۱- میانگین دو مولفه موردنظر.

Table 1- Average of the two desired components.

نرم‌افزار	سهولت استفاده	قابلیت اطمینان
GAMS	42.85	23.55
WinQSB	35.95	20.7
لیندو	38.15	21.7
لینگو	40.9	23.05
تورا	33.85	21.1

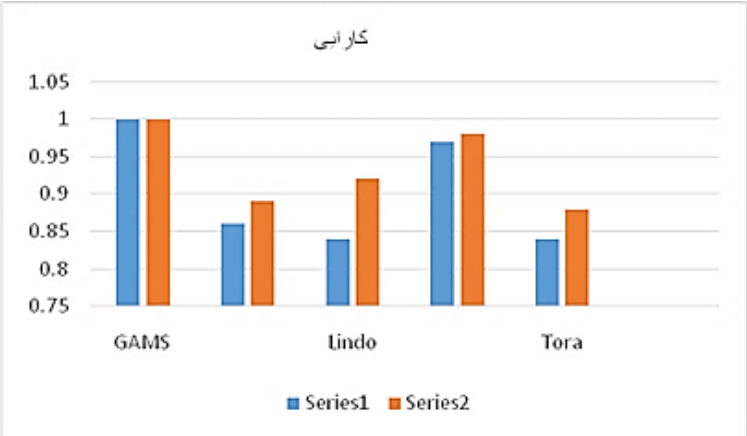
با توجه به اینکه این معیارها به‌عنوان خروجی در نظر گرفته‌شده است، با اجرای **مدل‌های (۳)** و **(۴)** که مدل‌های بدون ورودی هستند، نتایج در **جدول ۲** نشان شده است.

جدول ۲- نتایج حاصل از ارزیابی با مدل (۳) و (۴).

Table 2- The results of evaluation with model (3) and (4).

نرم‌افزار	مدل (۳)	مدل (۴)
GAMS	1	1
WinQSB	0.89	0.86
لیندو	0.92	0.84
لینگو	0.98	0.97
تورا	0.88	0.84
میانگین	0.93	0.90
واریانس	0.003	0.006

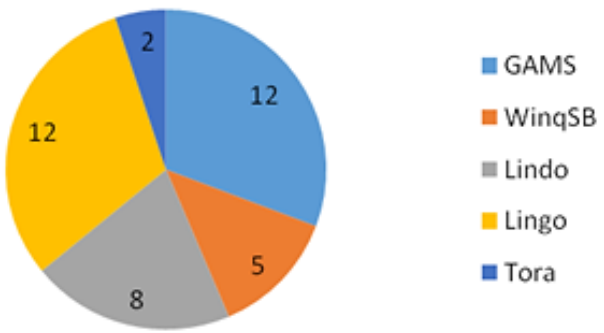
ستون دوم نتایج حاصل از اجرای **مدل (۳)** است. بنا به **جدول ۲**، نرم‌افزار *GAMS* در رتبه اول قرار دارد و لینگو با رتبه ۰/۹۸ در رتبه دوم قرار دارد. نتایج ستون سوم که حاصل از ارزیابی با **مدل (۴)** است نیز دلالت بر کارایی نرم‌افزار *GAMS* دارد و مانند نرم‌افزار لینگو در مرتبه دوم قرار دارد. نکته قابل تامل اینکه اختلاف بین دو نرم‌افزار گمز و لینگو به‌شدت اندک است. در **شکل ۱** اعداد حاصل از کارایی با اجرای دو **مدل (۳)** و **(۴)** باهم مقایسه شده‌اند. هر دو ارزیابی نرم‌افزار *GAMS* را در رتبه اول نشان می‌دهد و با اختلاف اندکی نرم‌افزار لینگو در رتبه دوم قرار دارد. با اجرای **مدل (۳)**، لیندو، *WinQSB* و تورا به ترتیب در جایگاه بعدی قرار دارد و با اجرای **مدل (۴)**، *WinQSB* جایگاه سوم را به خود اختصاص داده است و دو نرم‌افزار لیندو و تورا هر دو در جایگاه آخر قرار می‌گیرند. از سطر آخر **جدول ۲** چنین برمی‌آید که واریانس کارایی در **مدل (۳)** کمتر از واریانس کارایی محاسبه‌شده با **مدل (۴)** است. البته با توجه به اعداد حاصل این نتیجه دور از ذهن نیست.



شکل ۱- مقایسه اعداد کارایی دو مدل (۳) و (۴).

Figure 1- Comparison of efficiency numbers of two models (3) and (4).

با استخراج داده‌های حاصل از بیست پرسشنامه، **شکل ۲** نمودار قطاعی هست که هم ترازی دو نرم‌افزار لینگو و *GAMS* را نشان می‌دهد.



شکل ۲- مقایسه نرم‌افزارها.

Figure 2- Software comparison.

اعداد مندرج در **شکل ۲**، درواقع تعداد نمرات استخراج‌شده بالای شصت را برای هر نرم‌افزار نشان می‌دهد. با توجه به اینکه هر پرسشنامه در مقیاس لیکرت و با اعداد از یک تا پنج مقیاس‌گذاری شده است، حداکثر نمره قابل‌قبول ۹۰ و حداقل آن ۱۸ است. در تمام پرسشنامه‌ها، نمرات بالای ۶۰ به‌عنوان حد مطلوبیت آن نرم‌افزار شمارش شده است. ازاین‌رو فراوانی مندرج در نمودار قطاعی، تعداد نمرات بالای شصت به ازای هر بیست نفر است. همان‌طور که **شکل ۲** نشان می‌دهد دو نرم‌افزار لینگو و *GAMS* مطلوبیت یکسانی دارند و بقیه سه نرم‌افزار جایگاه بعدی را به خود اختصاص داده‌اند.

۴- نتیجه‌گیری

با پیشرفت الگوریتم‌ها و کدهای کامپیوتری و رایج شدن استفاده از آن در هر زمینه‌ای از تجارت و صنعت گرفته تا تفریح و سرگرمی، مساله ارزیابی نرم‌افزارها اهمیت ویژه‌ای پیدا می‌کند. دو معیار مهم قابلیت استفاده و قابلیت اطمینان در ارزیابی نرم‌افزارها در این ارزیابی موردنظر است. یکی از مسایل مهم در ارزیابی این دو فاکتور مهم انتخاب روش درست ارزیابی است که بتواند اطلاعاتی درست و مفید متناسب با هزینه و زمان صرف شده برای آن در اختیار تولیدکنندگان نرم‌افزار قرار دهد.

در این مقاله، روش ارزیابی بر اساس پرسشنامه استاندارد ارزیابی نرم‌افزارها به‌عنوان روش متداول مورداستفاده قرار گرفته است. با در اختیار داشتن نمونه تصادفی بیست‌تایی، داده‌ها استخراج‌شده و با استفاده از مدل‌های تحلیل پوششی داده‌ها بدون ورودی، پنج نرم‌افزار موردنظر در این مطالعه مورد ارزیابی قرار گرفتند. بررسی این پژوهش حاکی از برتری نرم‌افزار گمز در بین پنج نرم‌افزار مورداستفاده با لحاظ کردن تنها این دو عامل هست. این ارزیابی با روش‌های دیگری همچون *AHP*[1] و یا سایر روش‌های تصمیم‌گیری و با انتخاب معیارهای بیشتر می‌تواند به‌عنوان پژوهش آتی نیز در نظر گرفته شود.

منابع

Akhtarian, S., & Yaqoubi, T. (2016). A generalized model for software reliability evaluation based on heterogeneous Poisson process. *Journal of statistical sciences-scientific research journal of the Iranian statistical association*, *10*(1), 1-20. (**In Persian**). http://jss.irstat.ir/article-1-250-fa.html

Hosseinzadeh, L. F., Divandari, A., Jahanshahlou G., Nikumram H., & Baranki, T. W. (2016). Evaluating the supervision of bank Mellat of Iran in different periods of time .*Application of data coverage analysis*, *3*(8), 1-8. (**In Persian**). https://www.sid.ir/paper/164726/fa

Tehrani, R., & Jamshidi, H. (2014). Analyzing the influence of information, system and user interface design factors on the loyalty of customers of websites according to the role of satisfaction and trust. *Research journal of information processing and management (former information science and technology)*, *30*(4), 1085-1106. (**In Persian**). https://www.sid.ir/paper/131023/fa

Mazni, F., & Vafadar, S. (2018). Evaluation of software usability: a comparative study of questionnaires and users' performance. *Information processing and management research paper*, *34*(4), 1849-1878. (**In Persian**). https://jipm.irandoc.ac.ir/article-1-4174-fa.html

Abdelaziz, T. M., Maatuk, A. M., & Rajab, F. (2016). An approach to improvement the usability in software products. *International journal of software engineering & applications (IJSEA)*, *7*(2), 11-18. http://dx.doi.org/10.5121/ijsea.2016.7202

Arvanitou, E. M., Ampatzoglou, A., Chatzigeorgiou, A., & Carver, J. C. (2021). Software engineering practices for scientific software development: a systematic mapping study. *Journal of systems and software*, *172*, 110848. https://doi.org/10.1016/j.jss.2020.110848

Banker, R. D., Charnes, A., & Cooper, W. W. (1984). Some models for estimating technical and scale inefficiencies in data envelopment analysis. *Management science*, *30*(9), 1078-1092. https://doi.org/10.1287/mnsc.30.9.1078

Cai, Y., & Wu, W. (2001). Synthetic financial evaluation by a method of combining DEA with AHP. *International transactions in operational research*, *8*(5), 603-609. https://doi.org/10.1111/1475-3995.00336

Charnes, A., Cooper, W. W., & Rhodes, E. (1978). Measuring the efficiency of decision making units. *European journal of operational research*, *2*(6), 429-444. https://doi.org/10.1016/0377-2217(78)90138-8

Cooper, W. W., Seiford, L. M., & Tone, K. (2006). Introduction to data envelopment analysis and its uses: with DEA-solver software and references. *Springer science & business media*. https://doi.org/10.1007/0-387-29122-9

Huang, Y. S., Chiu, K. C., & Chen, W. M. (2022). A software reliability growth model for imperfect debugging. *Journal of systems and software*, *188*, 111267. https://doi.org/10.1016/j.jss.2022.111267

Ivanov, V., Reznik, A., & Succi, G. (2018). Comparing the reliability of software systems: a case study on mobile operating systems. *Information sciences*, *423*, 398-411. https://doi.org/10.1016/j.ins.2017.08.079

Juristo, N., Moreno, A. M., & Sanchez-Segura, M. I. (2007). Analysing the impact of usability on software design. *Journal of systems and software*, *80*(9), 1506-1516. https://doi.org/10.1016/j.jss.2007.01.006

Kumar, K., Kumar, S., Singh, L. K., & Mishra, A. (2022). Predicting reliability of software in industrial systems using a Petri net based approach: a case study on a safety system used in nuclear power plant. *Information and software technology*, 146, 106895. https://doi.org/10.1016/j.infsof.2022.106895

Liu, W. B., Zhang, D. Q., Meng, W., Li, X. X., & Xu, F. (2011). A study of DEA models without explicit inputs. *Omega*, *39*(5), 472-480. https://doi.org/10.1016/j.omega.2010.10.005

Lin, J. S., Huang, C. Y., & Fang, C. C. (2022). Analysis and assessment of software reliability modeling with preemptive priority queueing policy. *Journal of systems and software*, *187*, 111249. https://doi.org/10.1016/j.jss.2022.111249

Paiva, D. M. B., Freire, A. P., & de Mattos Fortes, R. P. (2021). Accessibility and software engineering processes: a systematic literature review. *Journal of systems and software*, *171*, 110819. https://doi.org/10.1016/j.jss.2020.110819

[1] Analytical Hierarchy Process (AHP)